

## Spectral Projected Gradient Method for the Procrustes Problem

J.B. FRANCISCO<sup>1</sup> and T. MARTINI<sup>2\*</sup>

Received on November 30, 2013 / Accepted on April 20, 2014

**ABSTRACT.** We study and analyze a nonmonotone globally convergent method for minimization on closed sets. This method is based on the ideas from trust-region and Levenberg-Marquardt methods. Thus, the subproblems consists in minimizing a quadratic model of the objective function subject to a given constraint set. We incorporate concepts of bidiagonalization and calculation of the SVD “with inaccuracy” to improve the performance of the algorithm, since the solution of the subproblem by traditional techniques, which is required in each iteration, is computationally expensive. Other feasible methods are mentioned, including a curvilinear search algorithm and a minimization along geodesics algorithm. Finally, we illustrate the numerical performance of the methods when applied to the Orthogonal Procrustes Problem.

**Keywords:** orthogonality constraints, nonmonotone algorithm, Orthogonal Procrustes Problem, Spectral Projected Gradient Method.

### 1 INTRODUCTION

In this paper, we consider the Weighted Orthogonal Procrustes Problem (WOPP). Then, given  $X \in \mathbb{R}^{m \times n}$ ,  $A \in \mathbb{R}^{p \times m}$ ,  $B \in \mathbb{R}^{p \times q}$  and  $C \in \mathbb{R}^{n \times q}$ , we address the following constrained optimization problem:

$$\begin{aligned} \min \quad & \frac{1}{2} \|AXC - B\|_F^2 \\ \text{s.t.} \quad & X^T X = I \end{aligned} \quad (1)$$

We also consider the problem when the objective function of (1) is replaced by  $\frac{1}{2} \|AX - B\|_F^2$ , called here as Orthogonal Procrustes Problem (OPP).

The Procrustes Problem belongs to the class of minimization problems restricted to the set of matrices with orthonormal columns, which often appear in important classes of optimization

---

\*Corresponding author: Tiara Martini.

<sup>1</sup>Department of Mathematics, Federal University of Santa Catarina, 88040-900 Florianópolis, SC, Brazil, grants 479729/2011-5. E-mail: juliano@mtm.ufsc.br

<sup>2</sup>Applied Mathematics Department, Mathematics, Statistics, and Cientific Computational Institute, State University of Campinas, 13083-970 Campinas, SP, Brazil. This author was supported by CAPES-PROF, Brazil. E-mail: tiaramartini@gmail.com

problems and, as well as the linear and quadratic constraints, has a special structure to be explored. As an example, we mention applications in body rigid movements [23], psychometry [17], multidimensional scaling [9] and Global Positioning System [4].

However, compute its solution is a costly task. In most cases, a solution is possible only computationally through an iterative method capable of dealing with non-stationary points and convex constraints, generating a sequence of feasible points converging to a local minimizer. Generate this feasible sequence carries a high computational cost, because it involves matrix reorthogonalization or generate points along geodesics. In the first case, some decomposition of the matrix, usually Singular Value Decomposition (SVD), is needed, and the second one is related to exponential of matrix or solving partial differential equations. Moreover, generally, these problems are classified as NP-hard. Recently it was proposed a free SVD and geodesic method that is very promising in terms of reduced computational time [24]. In the algorithms section we will talk more about these methods.

The aim of this study is to numerically compare the performance of these methods and improve the performance of the algorithm proposed in [13]. For this purpose we incorporate an alternative approach for calculating the SVD and a bidiagonalization strategy. With the first, our goal is to reduce the computational effort involved in this process, and with the second, we aim to solve a smaller problem, equivalent to the original.

This paper is organized as follows. In Section 2 we present the optimality conditions of the Procrustes Problem. In Section 3 we comment the methods studied. Section 4 is dedicated to the numerical results. Finally, the conclusions are commented in the last section.

Notation: The gradient of  $f$  will be denoted by  $g$  and the Hessian matrix by  $H$ , that is,  $g(X)_{ij} = \frac{\partial f}{\partial X_{ij}}(X)$  and  $H(X)_{ij,kl} = \frac{\partial^2 f}{\partial X_{ij} \partial X_{kl}}(X)$ . Given  $A \in \mathbb{R}^{n \times n}$ ,  $tr(A)$  means the trace of matrix  $A$ , and  $R(A)$  denote the column space of  $A$ .

## 2 THE ORTHOGONAL PROCRUSTES PROBLEM

The Procrustes Problem aims at analyzing the process of preserving a transformation to a set of forms. This process involves rotation and scaling in a certain data set, in order to move them to a common frame of reference where they can be compared. The known data are arranged in arrays and the rotation transformation to be determined is a matrix with orthonormal columns.

According to Elden & Park [11], the OPP problem with  $m = n$  is said to be balanced and the case where  $n < m$  is called unbalanced. For the balanced case, we have a closed formula for the solution, namely  $X_* = UV^T$ , where  $A^T B = U \Sigma V^T$  is the reduced SVD decomposition of  $A^T B$ . For unbalanced OPP and the WOPP cases, the solution is found through iterative methods, and therefore is not necessarily the global minimum.

Let  $f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$  defined as  $f(X) = \frac{1}{2} \|AXC - B\|_F^2$  and  $h : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{n \times n}$ ,  $m \geq n$ , defined by  $h(X) = X^T X - I$ , and consider  $\Omega = \{X \in \mathbb{R}^{m \times n} : h(X) = 0\}$ .

Note that  $f$  is a function of class  $C^2$  and  $g(X) = A^T(AXC - B)C^T \in \mathbb{R}^{m \times n}$ . Furthermore, as the Frobenius norm  $\|X\|_F = \sqrt{n}$ ,  $\forall X \in \Omega$ , it follows that  $\Omega$  is compact and  $g$  is Lipschitz continuous at  $\Omega$ .

The next theorem provides equivalent conditions for a matrix  $X \in \Omega$  to be a stationary point of the problem (1), which greatly simplifies the verification of stationary point.

**Theorem 1.** *Let  $X \in \Omega$ . Then  $X$  satisfies the KKT Conditions of (1) if, and only if,  $g(X) \in R(X)$  and  $X^T g(X)$  is a symmetric matrix.*

**Proof.** Note that the Lagrangian function of (1) is

$$l(X, \Theta) = f(X) + \text{tr}(\Theta(X^T X - I)),$$

where  $\Theta \in \mathbb{R}^{n \times n}$  contains the Lagrange multipliers of the problem. It follows that the KKT Conditions of (1) are:

$$g(X) + X(\Theta + \Theta^T) = 0 \text{ and } X^T X = I. \quad (2)$$

Assume that  $X$  satisfies (2). Then  $g(X) = -X(\Theta + \Theta^T)$ , i.e.,  $g(X) \in R(X)$ . Furthermore, as  $XX^T$  is the projection matrix on  $R(X)$ , so  $X^T g(X) = -(\Theta + \Theta^T)$ . Therefore,  $X^T g(X)$  is symmetric.

Now, considers that  $g(X) \in R(X)$  and  $X^T g(X)$  is symmetric. Define  $\Lambda = -\frac{X^T g(X)}{2}$ , so  $g(X) + X(\Lambda + \Lambda^T) = 0$ , i.e.,  $X$  satisfies (2).  $\square$

### 3 ALGORITHMS

The current section presents a special case of minimization in closed sets method proposed in [13] and briefly describe the methods of minimization along geodesic [1] and curvilinear search [24].

#### 3.1 Spectral Projected Gradient Method

The algorithm of [13] is based on trust region methods [20] and on the ideas of the Levenberg-Marquardt method [21], generating new iterates through the minimization of a quadratic model of the function around the current iterate. We will apply a particular case of this method to solve the problem (1).

Consider the constants  $0 < \sigma_{\min} < \sigma_{\max} < \infty$ ,  $L_u \geq L_f^1$ , for each  $k = 0, 1, 2, \dots$ , choose  $\sigma_{spc}^k \in [\sigma_{\min}, \sigma_{\max}]$ , and set

$$\sigma_{\rho}^k = \begin{cases} \frac{\sigma_{spc}^k}{2}, & \text{se } 0 < \rho < L_u \\ L_u, & \text{otherwise} \end{cases}, \quad (3)$$

where,  $\rho > 0$  is a regularization parameter.

---

<sup>1</sup> $L_f$  is the Lipschitz constant associated with the gradient of  $f$

Thus, the quadratic model  $Q_\rho^k : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$  is given by:

$$Q_\rho^k(X) = \text{tr}(g(X_k)^T(X - X_k)) + \frac{\sigma_\rho^k + \rho}{2} \|X - X_k\|_F^2.$$

For the parameter  $\sigma_{spc}^k$  let us choose, whenever possible, the Barzilai-Borwein parameter [3]. Thus, for two consecutive iterates,  $X_{k-1}$  and  $X_k$ , it follows that

$$\sigma_{bb}^k = \frac{\text{tr}((g(X_k)^T - g(X_{k-1})^T)(X_k - X_{k-1}))}{\|X_k - X_{k-1}\|_F^2},$$

and define

$$\sigma_{spc}^k = \begin{cases} 1, & \text{for } k = 0 \\ \min\{\max\{\sigma_{bb}^k, \sigma_{\min}\}, \sigma_{\max}\}, & \text{for } k \geq 1 \end{cases} \quad (4)$$

With these choices, the algorithm studied is a variation of the Nonmonotone Spectral Projected Gradient Method [5] for minimization with orthogonal constraints, and the solution of the sub-problem

$$\begin{aligned} \min \quad & Q_\rho^k(X) \\ \text{s.t.} \quad & X^T X = I \\ & X \in \mathbb{R}^{m \times n} \end{aligned}$$

is  $X^* = UV^T$ , where  $U \Sigma V^T$  is the reduced SVD decomposition of the matrix

$$W_k = X_k - \frac{1}{\rho + \sigma_\rho^k} g(X_k).$$

We mention that the nonmonotonic line search is based on that devised in [15] and is presented in the following definition.

**Definition 1.** Let  $\{x_k\}_{k \in \mathbb{N}}$  a sequence defined by  $x_{k+1} = x_k + t_k d_k$ ,  $d_k \neq 0$ ,  $t_k \in (0, 1)$ ,  $\gamma \in (0, 1)$ ,  $M \in \mathbb{Z}^+$  e  $m(0) = 0$ . For each  $k \geq 1$ , choose  $m(k)$  recursively by  $0 \leq m(k) \leq \min\{m(k-1) + 1, M\}$ , and set  $f(x_{l(k)}) = \max\{f(x_{k-j}) : 0 \leq j \leq m(k)\}$ . We say that  $x_{k+1}$  satisfies **Nonmonotone Armijo condition** if  $f(x_{k+1}) \leq f(x_{l(k)}) + \gamma t_k \nabla f(x_k)^T d_k$ .

Thus, we can now establish the algorithm for the problem (1).

The following theorem ensures the global convergence of the iterates sequence to stationary points and its proof can be found in [13].

**Theorem 2.** Let  $X$  be an accumulation point of the sequence generated by Algorithm 1. Then  $X$  is a stationary point of the problem (1), which is not a point of local maximum. In addition, if the set of stationary points of the problem is finite, then the sequence converges.

Algorithm 1: Projected Gradient for minimization on Stiefel manifolds (PGST).

Let  $X_0 \in \Omega$ ,  $M \in \mathbb{Z}^+$ ,  $\eta \in (0, 1]$ ,  $\beta_1 \in (0, \frac{1}{2}]$ ,  $0 < \sigma_{\min} < \sigma_{\max} < \infty$  and  $1 < \xi_1 \leq \xi_2 < \infty$ .

Set  $k = 0$  and  $m(0) = 0$ .

**Step 1.** Compute  $g(X_k)$ ,  $f(X_{l(k)})$  and set  $\rho = \frac{\sigma_{\rho c}^k}{2}$  as (4);

**Step 2.** Set  $\sigma_\rho^k$  defined in (3) and compute the reduced SVD of  $W_k$ :  $W_k = U\Sigma V^T$ ;

**Step 3.** Define  $\bar{X}_\rho^k = UV^T$  and find  $X_\rho^k \in \Omega$  such that  $Q_\rho^k(X_\rho^k) \leq \eta Q_\rho^k(\bar{X}_\rho^k)$ . If  $Q_\rho^k(X_\rho^k) = 0$ , declare  $X_k$  stationary point;

**Step 4.** Define

$$\Psi_k(X) = \text{tr}(g(X_k)^T(X - X_k)) + \frac{\sigma_\rho^k}{2} \|X - X_k\|_F^2.$$

If

$$f(X_\rho^k) \leq f(X_{l(k)}) + \beta_1 \Psi_k(X_\rho^k),$$

define  $\rho^k = \rho$ ,  $X_{k+1} = X_\rho^k$ , set  $k = k + 1$  and return to Step 1.

Else, choose  $\rho_{new} \in [\xi_1\rho, \xi_2\rho]$ , set  $\rho = \rho_{new}$  and return to Step 2.

### 3.2 Minimization method through geodesic

The minimization method through geodesic [1] is a modification of Newton's method for minimization in Stiefel and Grassmann manifolds that takes advantage of those geometries. In problems of unconstrained minimization, Newton's method consists of generating new iterates by subtracting of the current iteration a multiple of the vector  $H^{-1}\nabla f$ , where  $H$  is the Hessian matrix of the function  $f$  evaluated on the current iterate. Arias, Edelman and Smith [1] replaced this subtraction restricting the function along a geodesic path in the manifold. The gradient remained as usual (tangent to the surface of restrictions), while the Hessian matrix is obtained from the second differential of the function restricted to geodesic.

Let  $F_{XX}(\Delta_1, \Delta_2) = \sum_{ij,kl} H(X)_{ij,kl}(\Delta_1)_{ij}(\Delta_2)_{kl}$ , where  $\Delta_1, \Delta_2 \in \mathbb{R}^{m \times n}$ , and  $F_{XX}(\Delta)$  the single tangent vector which satisfies

$$F_{XX}(\Delta, Z) = \langle F_{XX}(\Delta), Z \rangle, \quad \forall Z \in \mathbb{R}^{m \times n}.$$

We present bellow, in general terms, the algorithm of [1], to be referred throughout this paper by `sg_min`. For more details see [1, 10].

In the experiments presented ahead we use the `dogleg` option of the package `sg_min`<sup>2</sup> in order to adjust the parameter of the linear search along the geodesic. In this case, instead of  $t = 1$ , we obtain a parameter  $t_k$  in (5) such that  $f(X(t_k)) < f(X)$ .

<sup>2</sup>The solver used is available in <http://web.mit.edu/~ripper/www/sgmin.html> (accessed in 29/01/2012)

Algorithm 2: Newton method for minimization on Stiefel manifolds (sg\_min).

Let  $X \in \Omega$ .

**Step 1.** Compute  $G = g(X) - Xg(X)^T X$ .

**Step 2.** Find  $\Delta$  such that  $(X^T \Delta)^T = -X^T \Delta$  and  $F_{XX}(\Delta) - X \text{skew}(g(X)^T \Delta) - \text{skew}(\Delta g(X)^T) - \frac{1}{2} \Pi \Delta X^T g(X) = -G$  where  $\text{skew}(X) = \frac{X-X^T}{2}$  and  $\Pi = I - XX^T$ .

**Step 3.** Move from  $X$  to  $X(1)$  in direction  $\Delta$  using the geodesic formula

$$X(t) = XM(t) + QN(t), \tag{5}$$

where  $Q$  comes from reduced QR decomposition of  $(I - XX^T)\Delta$ ,  $M(t)$  and  $N(t)$  are matrices given by the equation

$$\begin{pmatrix} M(t) \\ N(t) \end{pmatrix} = \exp t \begin{pmatrix} X^T \Delta & -R^T \\ R & 0 \end{pmatrix} \begin{pmatrix} I_n \\ 0 \end{pmatrix},$$

and return to Step 1.

### 3.3 Curvilinear search method

The method developed by [24] is a curvilinear search method over a manifold, which constructs a sequence of feasible iterates  $X_k$  as it follows: for each  $k \in \mathbb{N}$ , set

$$W_k = g(X_k)X_k^T - X_k g(X_k)^T \tag{6}$$

and

$$X_{k+1} = Y(t_k) = \left( I + \frac{t_k}{2} W_k \right)^{-1} \left( I - \frac{t_k}{2} W_k \right) X_k.$$

The step of the linear search under the curve  $Y$  is defined by the Barzilai-Borwein parameter [3]. To ensure global convergence of the method, Wen and Yin [24] adopted the strategy of nonmonotony of Zhang & Hager [26].

In the experiments we used the solver<sup>3</sup> based on Algorithm 2 of [24]. Below we present the algorithm that, throughout this paper, will be referred OptStiefel.

## 4 COMPUTATIONAL EXPERIMENTS

Below, we present each problem and we comment the obtained results when comparing the algorithms.

---

<sup>3</sup>The solver used is available in <http://optman.blogs.rice.edu/> (accessed in 29/01/2012)

Algorithm 3: Curvilinear search method with Barzilai-Borwein (OptStiefel).

Let  $X_0 \in \Omega$ ,  $t_0 > 0$ ,  $0 < t_m < t_M$ ,  $\rho_1, \delta, \eta, \varepsilon \in (0, 1)$ .

Set  $k = 0$ .

**Step 1.** Compute  $f(X_k)$ ,  $g(X_k)$  and define  $W_k$  as (6). If  $\|g(X_k)\| < \varepsilon$ , stop. Else, go to the next step.

**Step 2.** Compute  $f(Y(t_k))$ . While  $f(Y(t_k)) \geq C_k + \rho_1 t_k \langle g(X_k), W_k X_k \rangle$ , set  $t_k = \delta t_k$ .

**Step 3.** Define  $X_{k+1} = Y(t_k)$ ,  $q_{k+1} = \eta q_{k+1}$ ,  $C_{k+1} = \frac{\eta q_k C_k + f(X_{k+1})}{q_{k+1}}$ .  
Choose  $t_{k+1} = \max\{\min\{t_{k+1}, t_M\}, t_m\}$ , set  $k = k + 1$  and return to Step 1.

#### 4.1 Numerical experiments and problems

The experiments were performed using Matlab 7.10 on a intel CORE i3-2310M processor, CPU 2.10 GHz with 400 Gb of HD and 4 Gb of Ram. The following values for the constants of Algorithm 1 are used:  $\beta_1 = 10^{-4}$ ,  $\xi_1 = \xi_2 = 5$ ,  $\sigma_{\min} = 10^{-10}$ ,  $L_u = L_f$ ,  $\sigma_{\max} = L_u$  and  $M = 7$ . It is worth emphasizing that the above parameters were selected after a sequence of tests. This selection was focused in the parameters that, on average, showed the best results. For the others methods we used the default parameters presents in the authors' implementations solvers.

We assume convergence when  $X_k^T g(X_k)$  is symmetric and

$$\|g(X_k) - X_k(X_k^T g(X_k))\|_F \leq 10^{-6},$$

limiting the execution time (TMAX) into 600 seconds.

For the numerical experiments we consider  $n = q$ ,  $p = m$ ,  $A = PSR^T$  and  $C = Q\Lambda Q^T$ , with  $P$  and  $R$  random orthogonal matrices,  $Q \in \mathbb{R}^{n \times n}$  Householder matrix,  $\Lambda \in \mathbb{R}^{n \times n}$  diagonal with elements uniformly distributed in the interval  $[\frac{1}{2}, 2]$  and  $S$  diagonal defined for each type of problem. As a starting point  $X_0$  we generate a random matrix satisfying  $X_0 \in \Omega$ . All random values were generated with the `randn` function of Matlab.

Against the exact solution of the problem, we create a known solution  $Q_* \in \mathbb{R}^{m \times n}$  (a random Householder matrix) by taking  $B = A Q_* C$ , and thus to monitor the behavior of iterates  $X_k$ .

The tested problems were taken from [25] and are described below.

**Problem 1.** The diagonal elements of  $S$  are generated by a normal distribution in the interval  $[10, 12]$ .

**Problem 2.** The diagonal of  $S$  is given by  $S_{ii} = i + 2r_i$ , where  $r_i$  are random numbers uniformly distributed in the interval  $[0, 1]$ .

**Problem 3.** Each diagonal element of  $S$  is generated as follows:  $S_{ii} = 1 + \frac{99(i-1)}{m+1} + 2r_i$ , with  $r_i$  uniformly distributed in the interval  $[0, 1]$ .

In the following tables, we denote the number of iterations by  $N_i$ , the number of function evaluations by  $N_e$ , the residual  $\|AX_kC - B\|_F^2$ , the norm of the error in the solution  $\|X_k - Q_*\|_F^2$  and the time (in seconds). For these values we show the minimal (min), maximum (max) end the average (mean) of the results achieved from ten simulations.

First of all, we will test the efficiency of the nonmonotone line search. Note that the matrix  $A$  in Problem 1, that contains very near singular values, is well conditioned. Therefore, the use of nonmonotony can not present a significant reduction in the number of linear search, as seen in Table 1. For matrices of Problems 2 and 3, the singular values of  $A$  become more distant from each other as  $m$  increases, leaving the matrix ill-conditioned. For these problems, the strategy of nonmonotony shows its potential as can be seen in Table 1 and Figure 1.

Table 1: A comparative between the monotone and nonmonotone case of PGST method.

		Problem 1, $m = 500$				Problem 2, $m = 100$				Problem 3, $m = 100$			
		$q = 20$		$q = 70$		$q = 10$		$q = 50$		$q=10$		$q=50$	
		$N_i$	$N_e$	$N_i$	$N_e$	$N_i$	$N_e$	$N_i$	$N_e$	$N_i$	$N_e$	$N_i$	$N_e$
$M = 0$	min	18	24	20	26	602	970	736	1203	475	766	747	1238
	mean	21	29	22	29	903	1488	1029	1695	755	1243	1013	1675
	max	23	32	24	33	1517	2544	1564	2597	1171	1966	1395	2348
$M = 7$	min	17	22	19	24	489	661	557	743	558	763	597	806
	mean	19	24	20	24	790	1092	1043	1449	817	1129	940	1298
	max	20	25	21	25	994	1394	1716	2425	1078	1499	1409	1958

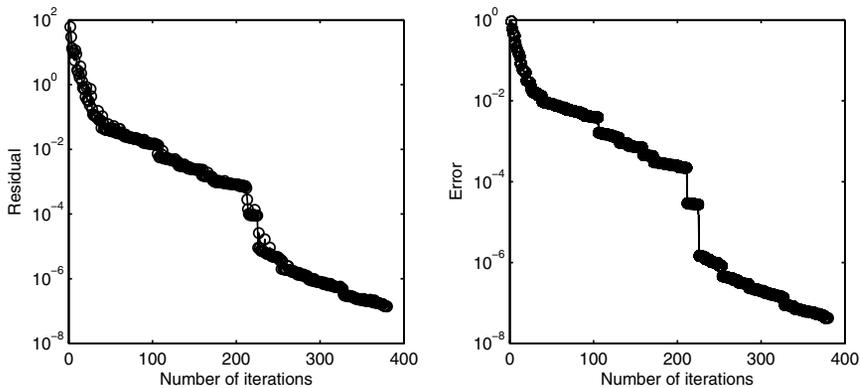


Figure 1: Behavior of PGST algorithm for Problem 2 with  $m = 50$  and  $q = 10$ . The y-axis is on a logarithmic scale.

Table 2 presents the results achieved for the problems. We can see that for well conditioned problems, PGST and OptStiefel algorithms showed similar performance, but in the presence of ill conditioning, the OptStiefel method shows a significant reduction from required values for convergence in comparison with other methods. In all cases examined, the method `sg_min` had a performance inferior to the others, and, in some situations, reached the maximum time allowed.

Table 2: Performance of the methods.

		Problem 1 with $m = 500$ and $q = 70$					Problem 1 with $m = 1000$ and $q = 100$				
Method		$N_i$	$N_e$	Time	Residual	Error	$N_i$	$N_e$	Time	Residual	Error
PGST	min	19	24	0.842	5.53e-09	3.31e-10	20	25	3.853	8.74e-09	6.82e-10
	mean	20	24	1.056	2.02e-08	1.41e-09	20	25	4.277	1.93e-08	1.43e-09
	max	21	25	1.700	3.31e-08	2.62e-09	21	26	6.084	3.71e-08	3.01e-09
OptStiefel	min	30	30	0.982	8.42e-09	6.05e-10	30	30	4.165	4.82e-09	2.54e-10
	mean	31	31	1.160	1.89e-08	1.33e-09	31	31	4.901	2.19e-08	1.64e-09
	max	34	34	1.372	3.47e-08	2.84e-09	33	33	5.272	3.75e-08	2.96e-09
sg_min	min	30	122	33.00	2.06e-08	1.53e-09	30	122	157.4	1.68e-08	1.28e-09
	mean	31	127	36.75	2.55e-08	1.88e-09	31	127	174.9	2.55e-08	1.88e-09
	max	32	130	40.31	3.16e-08	2.41e-09	32	130	197.3	3.46e-08	2.67e-09
		Problem 2 with $m = 100$ and $q = 50$					Problem 2 with $m = 500$ and $q = 20$				
PGST	min	557	743	4.695	4.02e-09	7.03e-10	7709	11212	87.09	1.00e-07	2.06e-08
	mean	1043	1449	11.16	1.12e-07	3.29e-08	11723	16972	153.3	1.27e-07	3.33e-08
	max	1716	2425	21.88	1.85e-07	7.12e-08	16780	24295	223.5	1.53e-07	4.74e-08
OptStiefel	min	435	453	1.700	3.41e-08	8.07e-09	1638	1722	14.38	1.74e-06	3.35e-07
	mean	559	591	2.280	1.19e-07	3.37e-08	2176	2285	20.57	7.32e-06	1.93e-06
	max	838	881	3.244	1.92e-07	5.75e-08	3008	3145	30.15	1.27e-05	3.91e-06
sg_min	med	—	—	TMAX	—	—	—	—	TMAX	—	—
		Problem 3 with $m = 500$ and $q = 20$					Problem 3 with $m = 1000$ and $q = 10$				
PGST	min	780	1082	8.221	5.15e-08	1.91e-08	781	1055	31.27	7.58e-08	3.19e-08
	mean	1150	1621	14.55	1.39e-07	5.21e-08	1457	2065	40.23	1.81e-07	7.83e-08
	max	2385	3434	25.13	2.73e-07	1.57e-07	2092	3042	55.39	2.50e-07	1.29e-07
OptStiefel	min	422	448	3.432	1.33e-07	4.21e-08	610	641	10.85	5.30e-07	1.76e-07
	mean	529	564	4.517	3.19e-07	1.16e-07	781	819	15.98	9.36e-07	3.52e-07
	max	732	764	6.942	5.93e-07	2.58e-07	1052	1092	20.99	1.46e-06	4.55e-07
sg_min	mean	—	—	TMAX	—	—	—	—	TMAX	—	—

#### 4.1.1 Using the Inexact SVD

Seeking to reduce the computational effort required to calculate the SVD in Algorithm 1, we propose to find this decomposition by an alternative algorithm [7], which builds recursively the SVD through calculation of the left and right dominant singular subspaces of dimension  $k$ ,

( $k \leq n$ ). This technique is appropriate for matrices where the number of rows is much larger than the number of columns.

The process begins with the QR decomposition of  $A_k$ , the matrix formed by the first  $k$  columns of  $A$ , which will be written in the form  $A_k V_k = Q_k R_k$ , with  $V_k = I_k$ . Then we proceed to update this decomposition, excluding the singular vector associated with the smallest singular value of the matrix of order  $m \times (k + i)$  considered and adding the next column of  $A$  to be analyzed. Thus, all columns of  $A$  are considered, generating only the singular vectors associated with the largest singular values of  $A$ . For more information about the method, including error analysis, see [7].

Table 3 displays the time, in seconds, spent by Algorithm 1 in its original form and when it uses the alternative SVD, the latter is referred to SISVD. For the cases  $q = 5$  and  $q = 10$  there is not a significant reduction, because the proportion between  $m$  and  $q$  is not enough to take advantage of the properties of the method. However, for  $q = 1$  we notice an improvement of 44%, on average, in performance.

Table 3: Comparative between PGST, SISVD and OptStiefel methods for problems 1, 2 and 3.

		Problem 1 $m = 5000$			Problem 2 $m = 500$			Problem 3 $m = 3000$		
		$q = 1$	$q = 5$	$q = 10$	$q = 1$	$q = 5$	$q = 10$	$q = 1$	$q = 5$	$q = 10$
PGST	min	1.076	4.290	6.037	8.299	21.20	31.69	39.31	132.1	192.1
	mean	1.301	5.745	7.126	13.50	52.18	106.6	66.30	212.4	297.9
	max	2.215	2.215	9.594	22.65	131.4	225.8	101.8	286.5	487.7
SISVD	min	0.592	4.258	5.709	4.664	21.57	45.22	19.34	129.0	190.6
	mean	0.809	5.589	7.781	7.477	49.76	105.7	37.18	207.7	293.7
	max	1.372	6.692	10.29	10.20	113.2	266.6	75.28	274.1	504.3
OptStiefel	min	0.811	4.633	5.912	2.246	4.914	7.550	20.45	67.90	84.00
	mean	1.054	6.108	7.296	4.871	8.966	14.03	26.57	92.68	118.3
	max	1.513	9.781	9.157	8.080	12.26	28.70	46.12	119.4	152.6

### 4.1.2 Using Golub-Kahan Bidiagonalization

With the purpose to solve large problems, we consider the problem OPP with  $q = 1$  and incorporate to Algorithm 1 the process of Golub-Kahan bidiagonalization presented in [22]. This process is based on the Lanczos algorithm for tridiagonalizing a symmetric matrix [19] and in Golub and Kahan bidiagonalization process [14]. In this case, we project the problem into a subspace, obtaining an equivalent problem, but with smaller dimension.

The Lanczos method consists of generating, from a symmetric matrix  $A \in \mathbb{R}^{n \times n}$  and a vector  $b \in \mathbb{R}^n$ , sequences of vectors  $\{v_k\}$  and of scalars  $\{\alpha_k\}$  and  $\{\beta_k\}$ , such that  $A$  is reduced to tridiagonal form, that is,  $V^T A V = T$  with  $T \in \mathbb{R}^{n \times n}$  tridiagonal and  $V \in \mathbb{R}^{n \times n}$ , satisfying

$V^T V = I$ . In turn, the Golub-Kahan bidiagonalization process applies the Lanczos process at a particular symmetrical system, namely

$$\begin{pmatrix} I & A \\ A^T & -\lambda^2 I \end{pmatrix} \begin{pmatrix} b - Ax \\ x \end{pmatrix} = \begin{pmatrix} b \\ 0 \end{pmatrix},$$

where  $\lambda \in \mathbb{R}$ .

So, after  $k$  steps of the Golub-Kahan algorithm we have  $AV_k = U_{k+1}B_k$  and  $A^T U_{k+1} = V_k B_k + \alpha_{k+1} v_{k+1} e_{k+1}^T$ , where  $B_k \in \mathbb{R}^{k+1 \times k}$  is bidiagonal,  $U_k = [u_1, u_2, \dots, u_k] \in \mathbb{R}^{m \times k}$  and  $V_k = [v_1, v_2, \dots, v_k] \in \mathbb{R}^{n \times k}$ , satisfies  $U_{k+1}^T U_{k+1} = I$  and  $V_k^T V_k = I$ .

Setting  $x = V_k y$ , we have  $\|Ax - b\|_F = \|B_k y - U_{k+1}^T b\|_F$ , with  $y = V_k^T x \in \mathbb{R}^k$ . Moreover,  $y^T y = 1$  if, and only if,  $x^T x = 1$ . Thus,

$$\begin{aligned} \min \quad & \|Ax - b\|_F \\ \text{s.t.} \quad & \|x\| = 1 \\ & x \in \text{span}\{V_k^1, \dots, V_k^k\}, \end{aligned}$$

where  $V_k^i$  is the  $i$ th-column of  $V_k$ , is equivalent to

$$\begin{aligned} \min \quad & \|B_k y - U_{k+1}^T b\|_F \\ \text{s.t.} \quad & \|y\| = 1. \end{aligned} \tag{7}$$

Then we apply the Algorithm 1 to problem (7), whose dimension is smaller than the size of the original problem. We will do the tests for matrices of Problem 1. The tests for the problems 2 and 3 were omitted, once the bidiagonalization process has not reduced the size of these problems, due to the fact that they are ill-conditioned.

The Algorithm 1 with the bidiagonalization process will be called SGKB. In Table 4 we expose the results obtained by applying SGKB, where the DSP is the size of the projected subspace, given by the number of iterations of the bidiagonalization process. The SGKB method showed promising results, since few steps of the bidiagonalization process were necessary and, therefore, the reduction in the size of the problem was significant. Additionally, the method quickly converges to a point very close to the solution.

## 5 CONCLUSIONS

We present a method to solve the Orthogonal Procrustes Problem with global convergence to stationary points. In order to verify the theoretical results and analyze the performance of the presented algorithm, we apply it in some particular cases of WOPP. The results were promising for the cases analyzed, in terms of reduced computational time, as compared with that obtained by usual methods. Therefore, the algorithm studied can be considered competitive when compared to methods already consolidated.

Table 4: Comparative between PGST and SGKB methods for OPP case of Problem 1.

		$m = 2000$						$m = 3000$					
Method		$N_i$	$N_e$	Time	Residual	Error	DSP	$N_i$	$N_e$	Time	Residual	Error	DSP
PGST	min	9	13	0.171	6.29e-09	5.28e-10	—	8	12	0.405	7.83e-09	6.58e-10	—
	mean	9	13	0.209	1.60e-08	1.35e-09	—	9	13	0.449	1.99e-08	1.71e-09	—
	max	9	13	0.218	2.90e-08	2.43e-09	—	9	13	0.546	3.93e-08	3.29e-09	—
SGKB	min	49	86	0.156	1.20e-08	1.01e-09	7	51	85	0.312	9.12e-09	8.99e-10	7
	mean	58	97	0.254	2.88e-08	2.58e-09	7	58	97	0.400	2.26e-08	2.10e-09	8
	max	67	110	0.358	4.20e-08	3.69e-09	8	64	107	0.530	3.15e-08	2.87e-09	8
		$m = 4000$						$m = 5000$					
PGST	min	9	13	0.670	5.90e-09	5.00e-10	—	8	12	0.967	5.87e-09	4.93e-10	—
	mean	9	13	0.726	1.21e-08	1.01e-09	—	8	13	1.282	2.06e-08	1.87e-09	—
	max	9	13	0.764	2.02e-08	1.70e-09	—	9	13	1.903	4.28e-08	3.99e-09	—
SGKB	min	52	86	0.530	1.57e-08	1.34e-09	7	52	86	0.717	1.36e-08	1.18e-09	7
	mean	57	95	0.606	2.22e-08	1.99e-09	7	56	93	0.928	2.70e-08	2.51e-09	7
	max	67	110	0.733	4.40e-08	3.93e-09	8	65	108	1.528	4.38e-08	4.06e-09	8

At the algorithm we incorporated bidiagonalization ideas and calculating the SVD in a alternatively way, which represented a significant improvement in performance when used for least-squares problems, or problems in which the number of rows of the matrix is much greater than the number of columns, respectively.

Additionally, we finalize this work by saying that the same approach here presented can be employed to minimize functionals over the symmetric matrices set. In that case, the ideas given in [16] for symmetric Procrustes Problems can be used to solve the subproblems that appear at Step 2 of Algorithm 1.

**ACKNOWLEDGMENTS**

The authors would like to thanks the anonymous referees for their carefully reading of the manuscript and for several constructive comments that improve significantly the presentation of this work.

**RESUMO.** Estudamos e analisamos um método globalmente convergente e não monótono para minimização em conjuntos fechados. Este método está baseado nas ideias dos métodos de região de confiança e Levenberg-Marquardt. Dessa maneira, os subproblemas consistem em minimizar um modelo quadrático da função objetivo sujeito a um conjunto de restrições. Incorporamos conceitos de bidiagonalização e de cálculo da SVD de maneira “inexata” buscando melhorar o desempenho do algoritmo, visto que a solução do subproblema por técnicas

tradicionalis, necessária em cada iteração, é computacionalmente muito cara. Outros métodos viáveis são citados, entre eles um método de busca curvilínea e um de minimização ao longo de geodésicas. Finalmente, ilustramos o desempenho numérico dos métodos quando aplicados ao Problema de Procrustes Ortogonal.

**Palavras-chave:** restrições de ortogonalidade, algoritmo não monótono, Método do Gradiente Projetado Espectral, Problema de Procrustes Ortogonal.

## REFERENCES

- [1] T.A. Arias, A. Edelman & S.T. Smith. “The geometry of algorithms with orthogonality constraints”. *SIAM Journal on Matrix Analysis and Applications*, **20**(2) (1998), 303–353.
- [2] A. Aspremont, L.E. Ghaoui, M. Jordan & G.R. Lanckriet. “A direct formulation for sparse pca using semidefinite programming”. *SIAM Review*, **49** (2007), 434–448.
- [3] J. Barzilai & J.M. Borwein. “Two point step size gradient methods”. *IMA Journal of Numerical Analysis*, **8** (1988), 141–148.
- [4] T. Bell. “Global Positioning System-Based Attitude Determination and the Orthogonal Procrustes Problem”. *Journal of guidance, control and dynamics*, **26**(5) (2003), 820–822.
- [5] E.G. Birgin, J.M. Martínez & M. Raydan. “Nonmonotone spectral projected gradient methods on convex sets”. *SIAM Journal on Optimization*, **10** (2000), 1196–1211.
- [6] A.W. Bojanczyk & A. Lutoborski. “The Procrustes Problem for Orthogonal Stiefel Matrices”. *SIAM Journal on Scientific Computing*, **21** (1998), 1291–1304.
- [7] Y. Chahlaoui, K. Gallivan & P. Van Dooren. “Recursive calculation of dominant singular subspaces”. *SIAM Journal on Matrix Analysis and Applications*, **25**(2) (1999), 445–463.
- [8] W.J. Cook, W.H. Cunningham, W.R. Pulleyblank & A. Schrijver. “Combinatorial optimization”. John Wiley & Sons, Canadá, (1997).
- [9] T.F. Cox & M.A.A. Cox. “Multidimensional scaling”. Chapman and Hall, Londres, (2000).
- [10] A. Edelman & R. Lippert. “Nonlinear Eigenvalue Problems with Orthogonality Constraints”. In *Templates for the Solution of Algebraic Eigenvalue Problems, A Practical Guide*, pp. 290–314, Philadelphia, (2000).
- [11] L. Eldén & H. Park. “A procrustes problem on Stiefel manifolds”. *Numerische Mathematik*, **82** (1999), 599–619.
- [12] J.B. Francisco, J.M. Martínez & L. Martínez. “Globally convergent trust-region methods for self-consistent field electronic structure calculations”. *Journal of Chemical Physics*, **121** (2004), 10863–10878.
- [13] J.B. Francisco & F.S. Viloche Bazán. “Nonmonotone algorithm for minimization on closed sets with application to minimization on Stiefel manifolds”. *Journal of Computational and Applied Mathematics*, **236** (10) (2012), 2717–2727.
- [14] G.H. Golub & W. Kahan. “Calculating the singular values and pseudoinverse of a matrix”. *SIAM Journal on Numerical Analysis*, **2** (1965), 205–224.

- [15] L. Grippo, F. Lampariello & S. Lucidi. “A nonmonotone line search technique for Newton’s method”. *SIAM Journal on Numerical Analysis*, **23**(4) (1986), 707–716.
- [16] N.J. Higham. “The symmetric procrustes problem”. *BIT Numerical Mathematics*, **28**(1) (1988), 133–143.
- [17] J.R. Hurley & R.B. Cattell. “The procrustes program: Producing direct rotation to test a hypothesized factor structure”. *Behavioral Science*, **7** (1962), 258–262.
- [18] E. Klerk, M. Laurent & P.A. Parrilo. “A PTAS for the minimization of polynomials of fixed degree over the simplex”. *Theoretical Computer Science*, pp. 210–225, (2006).
- [19] C. Lanczos. “An iteration method for the solution of the eigenvalue problem of linear differential and integral operators”. *Journal of Research of the National Bureau of Standards*, **45**(4) (1950), 255–282.
- [20] J.M. Martínez & S.A. Santos. “A trust region strategy for minimization on arbitrary domains”. *Mathematical Programming*, **68** (1995), 267–302.
- [21] J. Nocedal & S.J. Wright. “Numerical Optimization”. Springer, New York, (2006).
- [22] C.C. Paige & M.A. Saunders. “LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares”. *ACM Transactions on Mathematical Software*, **8**(1) (1982), 43–71.
- [23] I. Söderkvist & P. Wedin. “On Condition Numbers and Algorithms for Determining a Rigid Body Movement”. *BIT*, **34** (1994), 424–436.
- [24] Z. Wen & W. Yin. “A feasible method for optimization with orthogonality constraints”. *Optimization Online*, pp. 1–30, (2010).
- [25] Z. Zhang & K. Du. “Successive projection method for solving the unbalanced procrustes problem”. *Science in China: Series A Mathematics*, **49**(7) (2006), 971–986.
- [26] H. Zhang & W.W. Hager. “A nonmonotone line search technique and its application to unconstrained optimization”. *SIAM Journal Optimization*, **14**(4) (2004), 1043–1056.